



## 【実用新案登録請求の範囲】

【請求項1】 実行するプロセスとそのプロセスが使用するキャッシュ・メモリのバンク番号との対応関係を表わすテーブルを含むメイン・メモリと、このメイン・メモリの一部のコピーを持つために用意され、複数のバンクから構成されたキャッシュ・メモリと、現在のプロセスが使用しているキャッシュ・バンク番号を保持するためのレジスタと、このメイン・メモリからインストラクション・ワードをフェッチし、それを実行する機能と共に、実行されるプログラムに対してキャッシュ・バンク番号を前記テーブルに設定する機能を有するプロセッサを具備し、プロセスが切り替わった場合のキャッシュ・ヒット率の低下を抑え得るようにしたことを特徴とするキャッシュ制御装置。

## 【図面の簡単な説明】

【図1】 本発明に係るキャッシュ制御装置の一実施例を示す要部構成図である。

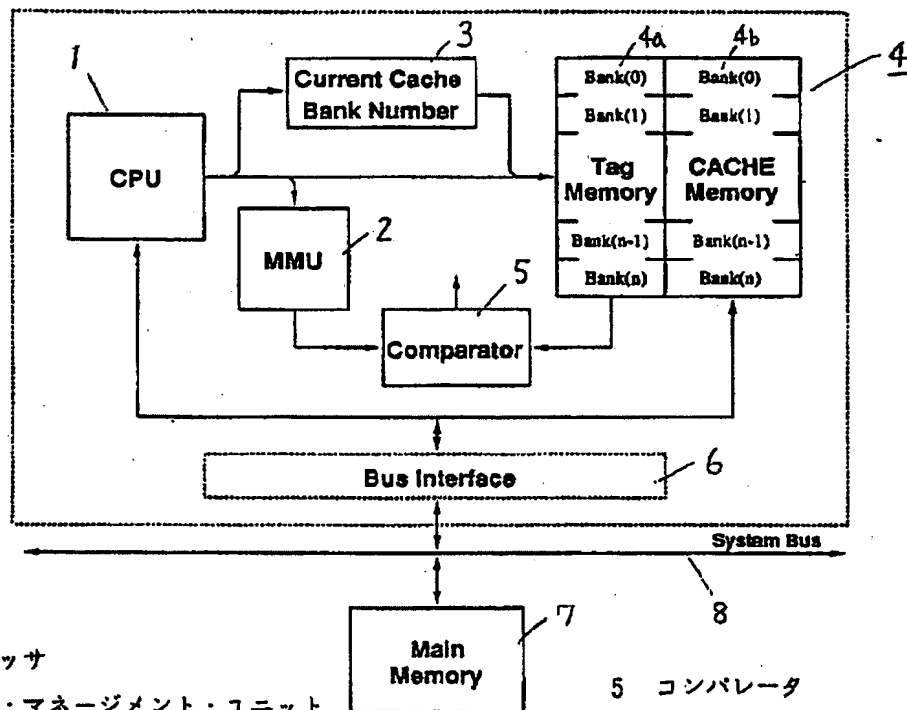
【図2】 プロセスIDとキャッシュIDとの対応関係を示す対応表である。

【図3】 動作説明のためのフローチャートである。

## 【符号の説明】

- 1 プロセッサ
- 2 メモリ・マネージメント・ユニット
- 3 レジスタ
- 4 キャッシュ・メモリ
- 5 コンパレータ
- 6 バス・インタフェース・ロジック
- 7 メイン・メモリ
- 8 システム・バス

【図1】



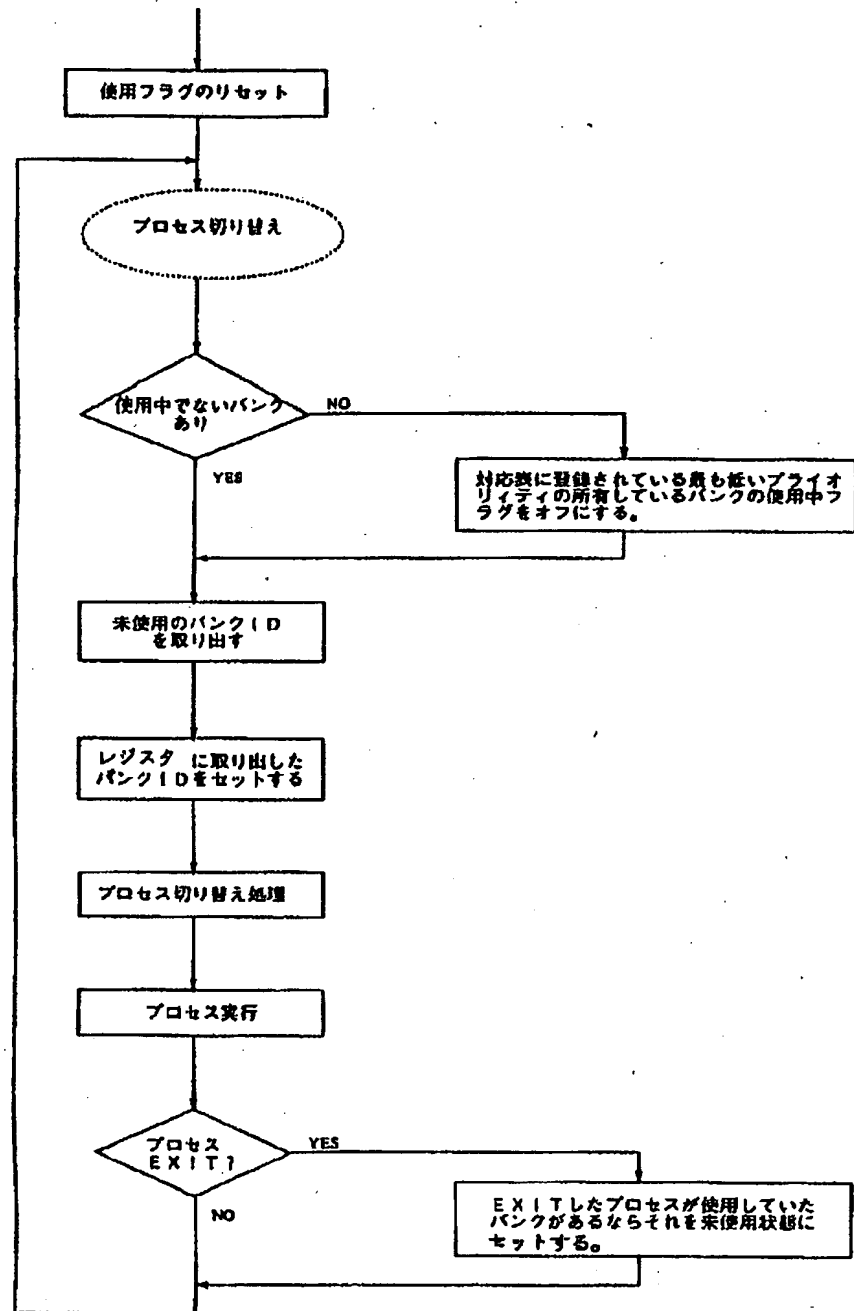
- 1 プロセッサ
- 2 メモリ・マネージメント・ユニット
- 3 レジスタ
- 4 キャッシュ・メモリ

- 5 コンパレータ
- 6 バス・インタフェース・ロジック
- 7 メイン・メモリ
- 8 システム・バス

【図2】

使用中 フラグ	プロセスID	キャッシュ・ バンクID
0	7	0
1	7	1
0	7	2
0	7	3

【図3】



## 【考案の詳細な説明】

## 【0001】

## 【産業上の利用分野】

本考案は、キャッシュ・メモリの制御方式の改善に関するものである。

## 【0002】

## 【従来の技術】

近年のコンピュータ・システムは高速に動作するマイクロプロセッサが使用されるようになってきた。このようなシステムにおいては、大容量のメモリを安価に構成できる低速のメモリ〔通常DRAM (Dynamic Random Access Memory) が使用される〕ではプロセッサのスピードに追従できず、プログラムの実行速度を改善するために小容量の高速のキャッシュ・メモリ〔通常SRAM (Static Random Access Memory) が使用される〕と共に使用するのが一般的である。

このキャッシュ・メモリは限定された個数のデータ・ワード、あるいはインストラクション・ワードを記憶しており、メモリの読み出し操作ごとにキャッシュ・メモリ中に必要なデータ、あるいはインストラクションが存在するかどうか調べられる。必要な情報がキャッシュ内に存在する場合（キャッシュ・ヒット）にはそのデータはキャッシュから読み出され、存在しない場合（キャッシュ・ミス）にはメイン・メモリから読み出される。そしてメイン・メモリから読み出されたデータ、あるいはインストラクションはキャッシュ・メモリ内の既存の情報と置き換えられる。

## 【0003】

## 【考案が解決しようとする課題】

ところで、このようなキャッシュ・ミスは、オペレーティング・システムの一つであるUNIX（登録商標）の環境下で行われるプロセス切り替えなどにおいては比較的顕著に現れ、その呼び出されたプロセスのデータおよびインストラクションでキャッシュ・メモリの内容が置き換えられる。そのためキャッシュ・ミスが多発し、更にこのプロセスを終了して元のプロセスに戻った時にも再びキャッシュ・ミスが生じるという問題があった。

本考案の目的は、このような点に鑑みてなされたもので、キャッシュ・メモリ

自体をバンク構造で構成することによってプロセス切り替えが起こった場合においてもキャッシュのヒット率を向上させ得るキャッシュ制御装置を実現しようとするものである。

#### 【0004】

##### 【課題を解決するための手段】

このような目的を達成するために本考案では、実行するプロセスとそのプロセスが使用するキャッシュ・メモリのバンク番号との対応関係を表わすテーブルを含むメイン・メモリと、このメイン・メモリの一部のコピーを持つために用意され、複数のバンクから構成されたキャッシュ・メモリと、現在のプロセスが使用しているキャッシュ・バンク番号を保持するためのレジスタと、このメイン・メモリからインストラクション・ワードをフェッチし、それを実行する機能と共に、実行されるプログラムに対してキャッシュ・バンク番号を前記テーブルに設定する機能を有するプロセッサを備える。

#### 【0005】

##### 【作用】

本考案では、キャッシュ・メモリをバンク構造にし、1つのプロセスごとにバンクを割り当てるようにする。

バンクがすべて使用中であれば、最も低いプライオリティのプロセスが使用しているキャッシュ・メモリのバンクを利用するように切り替える。

#### 【0006】

##### 【実施例】

以下図面を用いて本考案を詳しく説明する。図1は本考案に係るキャッシュ制御装置の一実施例を示す要部構成図である。図において、1はメモリからインストラクション・ワードをフェッチし、それを実行するためのプロセッサである。2はプロセッサ1から出力された論理アドレスを物理アドレスに変換するためのメモリ・マネージメント・ユニットである。3は現在の有効なキャッシュ・メモリのバンクを示すためのレジスタであり、プロセッサ1で実行されるインストラクションによりバンク番号が指定される。4はメモリの一部のコピーを持つためのキャッシュ・メモリであり、複数のバンクから構成され、タグ・メモリ4aと

キャッシュ・メモリ4bとから構成されている。プロセッサ1は、ある一時期においては、レジスタ3によって示されている1つのバンクからしかメモリ・ワードの参照を行わないようになっている。

#### 【0007】

5はメモリ・マネージメント・ユニット2で変換された物理アドレスとキャッシュ・メモリ4のタグ・メモリ4aから読み出された物理アドレス値とを比較するためのコンパレータである。この2つの値が一致する時はメイン・メモリ7へのアクセスが行われなくなっている。

6はシステム・バス8上に接続されたリソースとの間でデータの授受を行うためのバス・インタフェース・ロジックである。

図2はプロセスIDとキャッシュIDとの対応関係およびキャッシュ・バンクが使用中か否かを示すフラグに関するテーブルであり、メイン・メモリ7に記憶されている。

#### 【0008】

このような構成における動作を次に説明する。本考案では、キャッシュ・メモリを複数のバンクで構成することによって、プロセッサで実行する各プロセスごとに専用のキャッシュ・メモリのバンクを割り当てるようにしてある。

図3のフローチャートを参照して動作を説明すれば次の通りである。

- ① あるプロセスが終了するとプロセッサ1はそのプロセスに対応して立っていたフラグをリセットする（実施例では、使用中のフラグ1を0にする）。
- ② その後プロセスの切り替えが行われると、プロセッサ1は対応表を検索して未使用バンクが存在するか否かをチェックし、未使用バンクがない場合には対応表に登録されている最も低いプライオリティのプロセスが使用するキャッシュ・バンクの使用フラグをオフにする。
- ③ 未使用のバンクIDを取り出し、レジスタ3にセットする。
- ④ 当該プロセスに切り替え、実行する。
- ⑤ 当該プロセスが終了すると、終了したプロセスが使用していたバンクを未使用状態にフラグをリセットする。終了したプロセスが使用していたバンクがない場合、および当該プロセスの終了でない場合は上記②の処理に戻る。

以上のような方式により、プロセッサで実行されるプログラムによりキャッシュ・バンク番号が設定され、それぞれのプロセスに専用に見えるキャッシュ・バンクを与えることによって、プロセスが切り替わった場合のキャッシュ・ヒット率の低下を抑えることができる。

【0009】

【考案の効果】

以上述べたように、本考案によれば、キャッシュ・メモリを複数のバンクで構成し、それぞれのプロセスに専用に見えるその1つのバンクを割り当てることにより、サブ・プロセスをコールした場合のキャッシュ・ヒット率の向上を期待することができる。

さらに、リアルタイム処理を行うプロセスのキャッシュ・バンクを固定にすると、通常処理の中でリアルタイム処理を効率よく実現することができる。